Parallel Algorithms for Collective Processes in High Intensity Rings

A.Shishlo, V.Danilov, J.Holmes SNS-project, ORNL, Oak Ridge, TN, USA

The beam dynamics of present and planned high intensity rings, such as PSR, Fermilab Booster, AGS Booster, SNS, and proton driver, is dominated by collective processes. The details of these processes are sufficiently complicated that a good understanding of the underlying physics will require careful computer modeling. The simulation of such processes, including space charge and wake forces, requires three-dimensional modeling of the beam self and wall interactions. In many cases, the resulting simulations will require tracking tens of millions of interacting particles for thousands of turns, which constitutes a legitimate high performance computing problem. There is little hope of carrying out such calculations in a single processor environment. In order to meet the need for credible simulations of collective processes in high intensity rings, we are developing, implementing in the ORBIT computer code, and applying, parallel algorithms for the calculation of these processes.

Parallelization of a particle tracking code is best carried out by identifying clearly the collective and single particle dynamics. The parallel algorithms must be tailored to the collective processes to be modeled, while the single particle processes are transparent. In addition to longitudinal and transverse space charge and wake forces, many of the diagnostic quantities of interest are collective, in the sense that their definitions and evaluation involves many or all of the particles. All of these processes have to be prallelized, but the first for consideration is the space charge interaction, which is the most important for the time of calculations.

The three-dimensional space charge (3DSC) algorithm developed for ORBIT have been already parallelized and implemented as a separate module into the UAL 1.0 library, which supports a parallel environment based on MPI. This same process is now being carried out for the ORBIT code. The 3DSC algorithm and its parallelization are described in detail, together with its implications for the parallelization of the other collective and single particle processes. We will also present results demonstrating the scaling of these algorithms with problem size and number of processors.